LEVEL II

12

AD A108685

R and D CENTER

LABORATORY

TECHNICAL REPORT

NO. 12588

DYNAMIC ANALYSIS AND DESIGN OF CONSTRAINED MECHANICAL SYSTEMS

Interim Report

12 June 1981

Contract No. DAAK30-78-C-0096

Edward J. Haug, Roger Wehage
and N.C. Barman
College of Engineering
The University of Iowa
University of Iowa Rept. No. 50
by Ronald R. Beck, Project Engineer, TACOM

Approved for public release; distribution unlimited.

U.S. ARMY TANK-AUTOMOTIVE COMMAND
RESEARCH AND DEVELOPMENT CENTER
Warren, Michigan 48090

NOTICES

The findings in this report are not to be construed as an official Department of the Army position.

Mention of any trade names or manufacturers in this report shall not be construed as advertising nor as an official endorsement of approval of such products or companies by the U.S. Government.

Destroy this report when it is no longer needed. Do not return it to originator.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER 12588	2. GOVT ACCESSION NO. AD-A108	3. RECIPIENT'S CATALOG NUMBER 685
4. TITLE (and Subtitle) Dynamic Analysis and Design of Constrained Mechanical Systems		5. TYPE OF REPORT & PERIOD COVERED Interim to Nov. 78 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Edward J. Haug, Roger Wehage & N.C. Barman University of Iowa Ronald R. Beck, TACOM		8. CONTRACT OR GRANT NUMBER(s) DAAK30-78-C-0096
9. PERFORMING ORGANIZATION NAME AND ADDRESS The University of Iowa College of Engineering Iowa City, IA 52242		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS US Army Tank-Automotive Command R&D Center Tank-Automotive Concepts Lab, DRSTA-ZSA Warren, MI 48090		12. REPORT DATE 12 June 1981 13. NUMBER OF PAGES 55
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Dynamics, Bodies, Response, Control, Interaction, Stability, Lagrangian Function, Matrices

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A method for formulating and automatically integrating the equations of motion of quite general constrained dynamic systems is presented. Design sensitivity analysis is also carried out using a state space method that has been used extensively in structural design optimization. Both dynamic analysis and design sensitivity analysis and optimization are shown to be well-suited to application of efficient sparse matrix computational methods. Numerical integration is carried out using a stiff numerical integration method that treats mixed systems

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

4/1916

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

of differential and algebraic equations. A computer code that implements the method of planar systems is outlined and a numerical example is treated. The dynamic response of a classical slider-crank is analyzed and its design is optimized.

ABSTRACT

A method of formulating and automatically integrating the equations of motion of quite general constrained dynamic systems is presented. Design sensitivity analysis is also carried out using a state space method that has been used extensively in structural design optimization. Both dynamic analysis and design sensitivity analysis and optimization are shown to be well-suited to application of efficient sparse matrix computational methods. Numerical integration is carried out using a stiff numerical integration method that treats mixed systems of differential and algebraic equations. A computer code that implements the method for planar systems is outlined and a numerical example is treated. The dynamic response of a classical slider-crank is analyzed and its design is optimized.

I. Introduction to Computational Dynamics

Computational methods for dynamic analysis of electronic circuits [1] and structures [2] have become well developed and user-oriented, a situation that is not enjoyed in the field of mechanical system dynamics. In most areas of mechanical design, classical Lagrangian methods of dynamic analysis are still used almost exclusively. It is the purpose of this paper to present an integrated method of constrained mechanical system dynamic analysis and design sensitivity analysis that is user oriented and capable of routine application to large scale systems.

Development of general purpose computed codes for dynamic analysis of mechanisms and mechanical systems are initiated in the mid-1960's and has made significant progress for certain classes of mechanical systems. A comprehensive survey of computer codes and the analytical techniques on which they are based has been presented by Paul [3]. He cites numerous planar (2-D) mechanism analysis codes, but only two three-dimensional (3-D) analysis codes. At the present time only the 3-D Integrated Mechanism Program (IMP) [4] and the 2-D Dynamic Response of Articulated Machinery (DRAM) [5] codes are well developed and user-oriented. However, they are both based on closed loop linkages or mechanisms, which is too restrictive a class for general use in mechanical system dynamic analysis and design. A general 3-dimensional Automatic Dynamic Analysis of Mechanical Systems (ADAMS) method [6] has been presented that is fundamentally better suited for dynamic analysis of systems that are not in closed loop configurations. However, no generally applicable computer code is available at the present time. The ADAMS modelling method is all the more attractive since it is well-suited for extension to design sensitivity analysis.

Before presenting the theory it is helpful to constrast two fundamentally different approaches to modelling linkages and mechanical systems that are presently in use. The loop closure method generates equations that require closure of each independent loop of the linkage. The resulting nonlinear equations are then differentiated to obtain the smallest possible number of independent equations of motion, in terms of a minimum number of system degrees of freedom. Thus, the loop closure method generates a small number of highly nonlinear equations that are solved with standard numerical integration methods. The constrained system modelling method, on the other hand, explicitly treats three degrees of freedom for each element of the system (in the plane). Algebraic equations prescribing constraints between the various bodies are then written and elementary forms of equations of motion for each body are written separately. The constraint equations prescribing assembly of the mechanism are adjoined to the equations of motion through use of Lagrange multipliers. Thus, one treats a large number of equations in many variables. These equations may be solved by an implicit numerical integration method [1] that iteratively solves a linear matrix equation. The saving grace of this technique is that the matrix that arises in the iterative solution process is sparse. That is, only three to ten percent of the elements of the matrix are different from zero.

It is the purpose of this paper to develop the theory of dynamic and design sensitivity analysis and to present a computer code called Dynamic Analysis and Design System (DADS) that implements the constrained system modelling method for planar systems. The same basic theory is applicable for three-dimensional systems, but due to analytical complexity it will be presented in a separate paper.

2. Constrained Equations of Planar Motion

For planar mechanical systems, constraints between elements are taken as friction free (workless) translational and rotational joints. Extensions to include constraints such as cams, prescribed functional relations, and intermittent motion are possible by incorporating provisions for non-standard elements that are supplied by the user. In addition to standard constraints, springs and dampers connecting any pair of points on different bodies of the system are included in the model. In addition to these standard force elements, allowance is made for external forcing functions.

Generalized Coordinates: In order to specify the configuration or state of a planar mechanical system, it is first necessary to define generalized coordiates that specify the location of each body. As shown in Fig. 2.1, let the x-y coordinate system be an inertial reference frame. Define a body fixed coordinate system $\xi_i - \eta_i$ embedded in body i, with its origin at the center of mass. One can now locate the rigid body i by specifying the global coordinates (x_i, y_i) of the origin of the body fixed coordinates and the angle ϕ_i of rotation of this sytem relative to the global coordinates.

Let p be a point on body i, as shown in Fig. 2.1. The coordinates of the point in the body fixed $\xi_i - \eta_i$ system are ξ_i^p and η_i^p. The same point may also be located by its global coordinates x_i^p and y_i^p. The relation between these coordinates is

$$\begin{bmatrix} x_i^p \\ y_i^p \end{bmatrix} = \begin{bmatrix} x_i \\ y_i \end{bmatrix} + E(\phi_i) \begin{bmatrix} \xi_i^p \\ \eta_i^p \end{bmatrix} \tag{2.1}$$

where $E(\phi_i)$ is the transformation matrix

$$E(\phi_i) = \begin{bmatrix} \cos \phi_i & -\sin \phi_i \\ \sin \phi_i & \cos \phi_i \end{bmatrix} \tag{2.2}$$

In terms of these generalized coordinates, one can write the kinetic energy of the i^{th} body as

$$KE_i = \frac{1}{2} m_i (\dot{x}_i^2 + \dot{y}_i^2) + \frac{1}{2} J_i \dot{\phi}_i^2 \tag{2.3}$$

where m_i is the mass of the i^{th} body and J_i is its centroidal polar moment of inertia.

Further, one can write the virtual work of all externally applied forces on body i as

$$\delta W_e = Q_{x_i} \delta x_i + Q_{y_i} \delta y_i + Q_{\phi_i} \delta \phi_i \tag{2.4}$$

The effect of all forces except the workless constraint forces between bodies is included on δW_e of Eq. 2.4.

Equations of Constraint: Figure 2.2 depicts two adjacent bodies i and j. The origins of their body fixed coordinate systems are located by the vectors $\bar{R}_i$ and $\bar{R}_j$ with respect to the inertial frame of reference. Let an arbitrary point p_{ij} on body i be located by $\bar{r}_{ij}$ and p_{ji} on body j be located by $\bar{r}_{ji}$. These points are in turn connected by a vector $\bar{r}_p$. One can write a vector equation beginning at the origin of the inertial reference frame and closing there, to obtain the vector relationship

$$\bar{R}_i + \bar{r}_{ij} + \bar{r}_p - \bar{r}_{ji} - \bar{R}_j = \bar{0} \tag{2.5}$$

The constraint equations for a revolute joint are now obtained by requiring that p_{ij} and p_{ji} coincide. Setting $\bar{r}_p = \bar{0}$ and writing Eq. 2.5 in component form, one has

$$\left.\begin{aligned} x_i + \xi_{ij} \cos \phi_i - \eta_{ij} \sin \phi_i - x_j - \xi_{ji} \cos \phi_j + \eta_{ji} \sin \phi_j = 0 \\ y_i + \xi_{ij} \sin \phi_i + \eta_{ij} \cos \phi_i - y_j - \xi_{ji} \sin \phi_j - \eta_{ji} \cos \phi_j = 0 \end{aligned}\right\} \quad (2.6)$$

For a translational joint shown in Fig. 2.3, let points p_{ij} and p_{ji} lie on some line parallel to the path of relative motion between the two bodies. In addition, let them be located such that $\bar{r}_{ij}$ and $\bar{r}_{ji}$ are perpendicular to this line and of nonzero magnitude. Successively forming the dot product of Eq. 2.5 with $\bar{r}_{ij}$ and $\bar{r}_{ji}$ and adding, one obtains the scalar equation

$$- \bar{r}_{ij} \cdot \bar{r}_{ji} + r_{ij}^2 + \bar{r}_{ij} \cdot (\bar{R}_i - \bar{R}_j) + \bar{r}_{ji} \cdot \bar{r}_{ij} - r_{ji}^2 + \bar{r}_{ji} \cdot (\bar{R}_i - \bar{R}_j) = 0 \quad (2.7)$$

which reduces to

$$\begin{aligned} &(\xi_{ij} \cos \phi_i - \eta_{ij} \sin \phi_i + \xi_{ji} \cos \phi_j - \eta_{ji} \sin \phi_j)(x_i - x_j) \\ &+ (\xi_{ij} \sin \phi_i + \eta_{ij} \cos \phi_i + \xi_{ji} \sin \phi_j + \eta_{ji} \cos \phi_j)(y_i - y_j) \\ &+ \xi_{ij}^2 + \eta_{ij}^2 - \xi_{ji}^2 - \eta_{ji}^2 = 0 \end{aligned} \quad (2.8)$$

A second scalar equation is obtained by noting the $\bar{r}_{ij} \times \bar{r}_{ji} = \bar{0}$. Expansion of the cross product yields only a z component, which must be zero. This is

$$(\xi_{ij} \cos \phi_i - \eta_{ij} \sin \phi_i)(\xi_{ji} \sin \phi_j + \eta_{ji} \cos \phi_j)$$

$$- (\xi_{ji} \cos \phi_j - \eta_{ji} \sin \phi_j)(\xi_{ij} \sin \phi_i + \eta_{ij} \cos \phi_i) = 0 \qquad (2.9)$$

Spring-Dampers: Since springs and dampers generally appear in pairs, they are incorporated into a single set of equations. If one or the other is absent, its effect is eliminated by setting that term to zero. The equations for spring-damper force and torque are

$$\bar{F}_{ij} = \left[k_{ij}(\ell_{ij} - \ell_{0_{ij}}) + c_{ij} v_{ij} + F_{0_{ij}}\right] \frac{1}{\ell_{ij}} \bar{R}_{s_{ij}} \qquad (2.10)$$

$$T_{ij} = k_{r_{ij}} (\phi_{ij} - \phi_{0_{ij}}) + c_{r_{ij}} \dot{\phi}_{ij} + T_{0_{ij}} \qquad (2.11)$$

where

$\bar{F}_{ij}$ is the resultant force vector $[F_{x_{ij}}, F_{y_{ij}}]^T$ in the spring-damper

$\bar{R}_{s_{ij}}$ is the vector $[\ell_{ij} \cos \alpha, \ell_{ij} \sin \alpha]^T$ between points S_{ij} and S_{ji} of a spring-damper connection on the two bodies, as in Fig. 2.4

T_{ij} is the torque acting on two bodies at a revolute joint

k_{ij} and $k_{r_{ij}}$ are elastic spring coeffients

c_{ij} and $c_{r_{ij}}$ are damping coefficients

$\ell_{0_{ij}}$ and $\phi_{0_{ij}}$ are the undeformed spring length and angular rotation in the revolute

ℓ_{ij} and ϕ_{ij} are the deformed spring length and angular rotation in the revolute, and v_{ij} and $\dot{\phi}_{ij}$ are the time derivatives of ℓ_{ij} and ϕ_{ij}

$F_{0_{ij}}$ and $T_{0_{ij}}$ are constant forces and torques applied along the spring and around the revolute joint between two bodies

From Fig. 2.4 a vector expression similar to Eq. 2.5 is written as

$$\bar{R}_i + \bar{r}_{s_{ij}} + \bar{R}_{s_{ij}} - \bar{r}_{s_{ji}} - \bar{R}_j = 0$$

or in component form

$$\bar{R}_{s_{ij}} = \begin{bmatrix} \ell_{ij} \cos \alpha \\ \ell_{ij} \sin \alpha \end{bmatrix} = -\begin{bmatrix} x_i \\ y_i \end{bmatrix} - \begin{bmatrix} \xi_{s_{ij}} \cos \phi_i - \eta_{s_{ij}} \sin \phi_i \\ \xi_{s_{ij}} \sin \phi_i + \eta_{s_{ij}} \cos \phi_i \end{bmatrix}$$

$$+ \begin{bmatrix} x_j \\ y_j \end{bmatrix} + \begin{bmatrix} \xi_{s_{ji}} \cos \phi_j - \eta_{s_{ji}} \sin \phi_j \\ \xi_{s_{ji}} \sin \phi_j + \eta_{s_{ji}} \cos \phi_j \end{bmatrix} \tag{2.12}$$

where α is the angle between $\bar{R}_{s_{ij}}$ and the inertial x axis. Equation 2.12 is used to obtain ℓ_{ij} by noting that

$$\ell_{ij} = [(\ell_{ij} \cos \alpha)^2 + (\ell_{ij} \sin \alpha)^2]^{1/2}$$

$$= [(-x_i - \xi_{s_{ij}} \cos \phi_i + \eta_{s_{ij}} \sin \phi_i + x_j + \xi_{s_{ji}} \cos \phi_j$$

$$- \eta_{s_{ji}} \sin \phi_j)^2 + (- y_i - \xi_{s_{ij}} \sin \phi_i - \eta_{s_{ij}} \cos \phi_i$$

$$+ y_j + \xi_{s_{ji}} \sin \phi_j + \eta_{s_{ji}} \cos \phi_j)^2]^{1/2} \tag{2.13}$$

Substituting the left side of Eq. 2.12 into Eq. 2.10, the following force expressions are obtained:

$$F_{x_{ij}} = |\bar{F}_{ij}| \cos \alpha \tag{2.14}$$

$$F_{y_{ij}} = |\bar{F}_{ij}| \sin \alpha \tag{2.15}$$

where cos α and sin α are obtained by dividing Eq. 2.12 by ℓ_{ij}. Finally, defining

$$v_{ij} = \dot{\ell}_{ij} \tag{2.16}$$

and transfering ℓ_{ij}, $F_{x_{ij}}$, $F_{y_{ij}}$, and v_{ij} to the right hand sides of Eqs. 2.13 to 2.16, one obtains equations in the form required by the numerical integration algorithm.

System Equations of Motion: With the kinetic energy given by Eq. 2.3 and the generalized forces associated with externally applied forces and spring-dampers, one can construct the equations of motion for the system. To allow for a unified development, let q^i denote the vector $[x_i, y_i, \phi_i]^T$ of generalized coordinates of body i. Denote the kinetic energy of body i as $KE_i(q^i)$, the generalized force on body i as $Q^i(q^i)$, and the equations of constraint between bodies i and j as $\Phi^{ij}(q^i, q^j) = 0$, $i < j$. Here the constraint numbering convention is that $i < j$ in order to preclude inclusion of the same set of constraint equations twice.

Presuming that the constraints are workless, one has the variational equation of motion [7]

$$\sum_i \frac{d}{dt}\left(\frac{\partial KE_i}{\partial \dot{q}^i}\right) \delta q^i - \sum_i Q^{i^T} \delta q^i = 0 \tag{2.17}$$

that must hold for all time and for all virtual displacements δq that are consistent with the constraints. This is

$$\frac{\partial \Phi^{ij}}{\partial q^i} \delta q^i + \frac{\partial \Phi^{ij}}{\partial q^j} \delta q^j = 0 , \quad i < j \tag{2.18}$$

By the Farkas Lemmas of optimization theory [9], there exist multipliers λ^{ij}, $i < j$, such that

$$\sum_i \frac{d}{dt}\left(\frac{\partial KE_i}{\partial \dot{q}^i}\right) \delta q^i - \sum_i Q^{i^T} \delta q^i + \sum_{i<j} \lambda^{ij^T} \left[\frac{\partial \Phi^{ij}}{\partial q^i} \delta q^i + \frac{\partial \Phi^{ij}}{\partial q^j} \delta q^j\right] = 0 \tag{2.19}$$

for all δq^i. By Eq. 2.3, $KE_i = \frac{1}{2} \dot{q}^{i^T} M^i \dot{q}^i$, where

$$M^i = \begin{bmatrix} m_i & 0 & 0 \\ 0 & m_i & 0 \\ 0 & 0 & J_i \end{bmatrix}$$

and Eq. 2.19 may be written in the form

$$\sum_i \left[\ddot{q}^{i^T} M^i - Q^{i^T} + \sum_{i<j} \lambda^{ij^T} \frac{\partial \Phi^{ij}}{\partial q^i} + \sum_{j<i} \lambda^{ji^T} \frac{\partial \Phi^{ji}}{\partial q^i}\right] \delta q^i = 0 \tag{2.20}$$

Since Eq. 2.20 must hold for arbitrary δq^i, this yields the equations of motion

$$M^i \ddot{q}^i - Q^i(q^i) + \sum_{i<j} \frac{\partial \Phi^{ij}}{\partial q^i} \lambda^{ij} + \sum_{j<i} \frac{\partial \Phi^{ji}}{\partial q^i} \lambda^{ji} = 0 , \quad i=1,\ldots,r \tag{2.21}$$

These differential equations and the algebraic constraint equations

$$\phi^{ij}(q^i,q^j) = 0 \ , \quad i < j \tag{2.22}$$

form the system equations of motion. Defining the generalized velocity vector

$$u^i = \dot{q}^i$$

one can replace the second order system of Eqs. 2.21 by the first order system

$$M^i\dot{u}^i - Q^i(q^i) + \sum_{i<j} \frac{\partial \phi^{ij}}{\partial q^i} \lambda^{ij} + \sum_{j<i} \frac{\partial \phi^{ji}}{\partial q^i} \lambda^{ji} = 0 \tag{2.23}$$

$$\dot{q}^i - u^i = 0 \tag{2.24}$$

The problem of dynamic analysis is thus reduced to solving the system of differential and algebraic equations of Eqs. 2.22 to 2.24, with initial conditions given that are consistent with the constraints of Eqs. 2.22. The solution variables are q^i and u^i, i=1,...,r and λ^{ij} for all $i < j$. Each of these variables is a function of time.

3. Integration of Constrained Equations of Motion

Numerical Integration of Differential Equations: In order to solve the differential equations of motion, numerical integration theory must be used to obtain a set of approximations that are suitable for digital computation. Integration of the equations of motion must be accomplished as asimultaneous solution of the algebraic and differential equations of Eqs. 2.22 to 2.24. Standard numerical methods are however designed to solve only systems of differential equations of the form

$$\dot{y} = f(y,t) \tag{3.1}$$

where y is an m-vector of unknowns and f is an m-vector of known functions. A modified approach is taken here that allows for the simultaneous solution of mixed algebraic and differential equations of the form

$$g(y,\dot{y},t) = 0 \tag{3.2}$$

where some components of $\dot{y}$ may not appear in any of the equations, or they may appear nonlinearly. Before introducing the method to be used, it is instructive to review the approach applied in solving Eq. 3.1.

The basic method of constructing approximate solutions [1] is to place a time grid t_i, i=1,... on the interval $[0,\tau]$, where $h_i = t_{i+1} - t_i$ is the grid spacing. One then approximates the solution y(t) of Eq. 3.1 on the time grid as $y^i \approx y(t_i)$.

The basic approximating equation for stiff differential equations [1], i.e. differential equations with widely split eigenvalues, is the Gear algorithm

$$y^{i+1} - h_i\beta_0\, \dot{y}^{i+1} = \sum_{j=1}^{k} \alpha_j\, y^{i-j+1} \tag{3.3}$$

where the constants β_0 and α_j, j=1,..., k, called Gear's coefficients, are determined so that Eq. 3.3 is exact for any polynomial solution of Eq. 3.1 of degree up to k. Gear's coefficients [1] also have the property that the algorithm tends to be stable, even for stiff differential equations.

The multistep formula that is used to solve Eq. 3.2 is derived from Eq. 3.3. One progresses from t_i to t_{i+1} by solving Eq. 3.3, together with

$$g(y^{i+1}, \dot{y}^{i+1}, t_{i+1}) = 0 \tag{3.4}$$

Linearization of Eq. 3.4 generates the Newton formula

$$\frac{\partial g^{(m)}}{\partial y}\, \Delta y^{(m)} + \frac{\partial g^{(m)}}{\partial \dot{y}}\, \Delta \dot{y}^{(m)} = -\, g^{(m)} \tag{3.5}$$

where $\Delta y^{(m)} = y^{(m+1)} - y^{(m)}$ and m represents the iteration number. The time step index i has been suppressed here for notational simplicity. Substitution of Eq. 3.5 into Eq. 3.3, noting that the summation term of Eq. 3.3 remains constant at each iteration, yields the corrector formulas

$$\left[\frac{1}{h\beta_0}\, \frac{\partial g^{(m)}}{\partial \dot{y}} + \frac{\partial g^{(m)}}{\partial y}\right] \Delta y^{(m)} = -\, g^{(m)} \tag{3.6}$$

$$\Delta \dot{y}^{(m)} = \frac{1}{h\beta_0}\, \Delta y^{(m)} \tag{3.7}$$

If Eq. 3.2 is of the form $P\dot{y} + f(y,t) = 0$, then Eq. 3.6 is of the form

$$\left[\frac{1}{h\beta_0} P + \frac{\partial f^{(m)}}{\partial y}\right]\Delta y^{(m)} = - g^{(m)} \tag{3.8}$$

The iterative corrector procedure is continued at each time step until all of the Newton differences $\Delta y^{(m)}$ are below a specified tolerance level. At each iteration y and $\dot{y}$ are updated as

$$y^{(m+1)} = y^{(m)} + \Delta y^{(m)}$$

$$\dot{y}^{(m+1)} = \dot{y}^{(m)} + \Delta \dot{y}^{(m)} = \dot{y}^{(m)} + \frac{1}{h\beta_0} \Delta y^{(m)}$$

It is interesting to note that every element of $\dot{y}$ is obtained at each time step, even though many elements of $\dot{y}$ may not appear in any of the equations.

Sparse Matrix Algebra: The sytem of nonlinear algebraic and differential equations defined in the previous paragraphs is very loosely coupled. For two reasons, no attempt is made to eliminate variables to obtain a smaller system of equations. First, the Jacobian matrix that is formed by linearization of the loosely coupled equations and used in iterative solution by Newton's method is sparse and can be very efficiently stored and decomposed. Second, the repetitive nature of the loosely coupled equations results in compact and efficient computer routines for evaluating the nonlinear equations and nonzero matrix entries. Recently developed sparse matrix algorithms [8] make both of these operations practical and desirable. It has been shown [8] that it is usually more efficient to solve large systems of sparse equations, rather than smaller systems with greater percentages of nonzero entries.

Consideration of matrix sparsity is important to the speed of computation in problems of dynamic system analysis [1]. When less than 30% of the matrix entries are nonzero, it is inefficient to store the matrix as a full two-dimensional array. Instead, only the nonzero entries are stored in compacted form. The method most commonly used for compacting the data is to store the row and column indices of each nonzero-valued entry in the matrix in two vectors I and J and the value in a third vector A. This is called "i-j" ordering.

Sparse matrix algorithms are most efficient when the nonzero matrix entries are stored in an organized manner. This usually implies that they be evaluated row by row or column by column. The previously mentioned repetitive matrix evaluation scheme is defeated by this requirement, since it usually results in the evaluation of small submatrices located at various positions throughout the matrix. To overcome this difficulty, a permutation vector is generated from the row and column vectors describing the nonzero-valued positions. As each matrix entry is generated, it is directed to a specific location in the "A" vector by a permutation index. At completion of the matrix evaluation, all entries are stored exactly as if they had been evaluated in column order.

A sparse matrix decomposition algorithm is then applied to the column ordered matrix and an LU factorization [1] is accomplished. Full pivoting is not achieved, but the algorithm chooses, from among the largest of the acceptable pivot elements, the pivot that results in a minimum number of fills in the resulting L and U matrices. This is important for efficient execution of the forward and backward substitution phases, since an increased number of fills destroys the original matrix sparsity and results in excessive computer time.

4. Elements of Design Sensitivity Analysis and Optimization

As shown in this section and developed in more detail in companion papers [9], the constrained dynamic system model of Section 2 and the implicit numerical integration formulation of Section 3 are ideally suited for efficient design sensitivity analysis. If the designer identifies a set of parameters $b = [b_1,\ldots,b_\zeta]^T$ that he uses to specify the system design, a natural question that arises is: How does a design variation $\delta b = [\delta b_1,\ldots,\delta b_\zeta]^T$ effect the dynamic response of the system? An even more important question is: What is the effect on dynamic response of a design variation δb that is required to be consistent with certain performance requirements? A method of design sensitivity analysis developed for related classes of problems [10] is employed here to answer these questions.

In order to address the design sensitivity analysis problem, a compact notation is required. Let the vectors $z(t) = [z_1(t),\ldots,z_m(t)]^T$ contain all generalized displacement and velocity coordinates; $\ell(t) = [\ell_1(t),\ell_2(t),\ldots]^T$ contain spring lengths, velocities and forces; and $\lambda(t) = [\lambda^{ij^T}(t),\ i < j]^T$ contain Lagrange multipliers that are associated with constraints (which determine reaction forces associated with constraints).

With this notation, the equations of motion are written in compact form as

$$\left.\begin{aligned} P(b)\dot{z} + f(t,z,\lambda,\ell,b) &= 0 \\ z(0) &= \nu(b) \end{aligned}\right\} \qquad (4.1)$$

The matrix P is made up of 6 x 6 matrices that are associated with

derivative terms in Eqs. 2.23 and 2.24 for each of the bodies; i.e., $P_i(b) = \text{diag}\,(m_i(b), m_i(b), J_i(b), 1, 1, 1)$. Similarly, the vector f^i is made up of the remaining terms in Eqs. 2.23 and 2.24; i.e.,

$$f^i = -\Bigg[\Big(Q_{x_i} - \sum_{i<j} \frac{\partial \phi^{ij}}{\partial x_i}\lambda^{ij} - \sum_{j<i} \frac{\partial \phi^{ji}}{\partial x_i}\lambda^{ji}\Big),$$

$$\Big(Q_{y_i} - \sum_{i<j} \frac{\partial \phi^{ij}}{\partial y_i}\lambda^{ij} - \sum_{j<1} \frac{\partial \phi^{ji}}{\partial y_i}\lambda^{ji}\Big),$$

$$\Big(Q_{\phi_i} - \sum_{i<j} \frac{\partial \phi^{ij}}{\partial \phi_i}\lambda^{ij} - \sum_{j<i} \frac{\partial \phi^{ji}}{\partial \phi_i}\lambda^{ji}\Big), u_i, v_i, \omega_i\Bigg]^T$$

The spring damper equations of Eqs. 2.13 to 2.16 in matrix form are

$$\left.\begin{aligned} &\bar{I}\dot{\ell} + \theta(z,\ell,b) = 0 \\ &\ell^{(0)}_{1+4j} = \bar{v}_{j+1}(b),\ j = 0,1,\ldots,s-1 \end{aligned}\right\} \tag{4.2}$$

where s is the number of spring damper pairs and the matrix $\bar{I}$ is diagonal and contains all zeros, except for ones on the diagonal corresponding to coefficients of $\dot{\ell}_{ij}$ in Eq. 2.16. The vector θ simply contains all other terms of Eqs. 2.13 to 2.16.

Finally, the set of kinematic constraints is written in vector form as

$$\Phi(z,b) = 0 \tag{4.3}$$

The dependence of Eqs. 4.1 to 4.3 on the design variable b represents the fact that b may include such parameters as spring and joint locations, spring and damper coefficients, masses, and moments of inertia.

Optimal Design Problem Formulation: For the purpose of this introductory discussion of design sensitivity analysis, the cost function to

be minimized and performance constraints to be satisfied are taken in integral form. The general design problem is then to minimize

$$\psi_0 = g_0(b) + \int_0^\tau L_0(t,z,\lambda,\ell,b)dt \tag{4.4}$$

subject to the constraints

$$\psi_i = g_i(b) + \int_0^\tau L_i(t,z,\lambda,\ell,b)dt \begin{cases} = 0, & i=1,\ldots,r' \\ \leq 0, & i=r'+1,\ldots,r \end{cases} \tag{4.5}$$

This form of integral constraint can be made to include constraints that must hold for all time; i.e.,

$$\eta(t,z\ (t),\lambda(t),\ell(t),b) \leq 0\ , \qquad 0 \leq t \leq \tau \tag{4.6}$$

The inequality of Eq. 4.6 is equivalent to the integral constraint

$$\int_0^\tau [\eta(t,z(t),\lambda(t),\ell(t),b) + |\eta(t,z(t),\lambda(t),\ell(t),b)|]dt = 0$$

which is of the form of Eq. 4.5. This technique has been used for a wide variety of constrained dynamic systems [11] with good success. It may be noted that constraints of the form of Eq. 4.6 may represent excursion limits, bounds on stress, path generation error bounds, and a multitude of other forms of meaningful design constraints.

Design Sensitivity Analysis: In order to determine the effect of a design variation δb, one must first note that the solution of Eqs. 4.1 to 4.3 will be changed by perturbations δz, $\delta \ell$, and $\delta \lambda$. Presuming the design problem is well posed, small perturbations δb in b will lead to small perturbations in the solution variables.

Consider a typical functional of Eq. 4.4 or 4.5 where the subscript has been deleted for notational simplicity. To first order, one has

$$\delta\psi = \frac{\partial g}{\partial b}\,\delta b + \int_0^\tau \left[\frac{\partial L}{\partial z}\,\delta z + \frac{\partial L}{\partial \lambda}\,\delta\lambda + \frac{\partial L}{\partial \ell}\,\delta\ell + \frac{\partial L}{\partial b}\,\delta b\right] dt \tag{4.7}$$

The objective is to eliminate explicit dependence of Eq. 4.7 on δz, $\delta\lambda$, and $\delta\ell$ and to write $\delta\psi$ explicitly in terms of δb. To do this, an effective technique is to introduce adjoint variables $\bar{\mu}$, μ', and μ^*, multiplying Eqs. 4.1 to 4.3 respectively, and integrate to obtain the identities

$$\left.\begin{aligned}
&\int_0^\tau \{\bar{\mu}^T[P(b)\dot{z} + f(t,z,\lambda,\ell,b)]\}dt = 0 \\
&\int_0^\tau \{\mu'^T[\bar{I}\dot{\ell} + \theta(z,\ell,b)]\}dt = 0 \\
&\int_0^\tau \{\mu^{*T}\Phi(z,b)\}dt = 0
\end{aligned}\right\} \tag{4.8}$$

Then integrate Eqs. 4.8 by parts and require that $\bar{\mu}$, μ', and μ^* satisfy the adjoint equations

$$\left.\begin{aligned}
&-P^T\dot{\bar{\mu}} + \frac{\partial f}{\partial z}^T\bar{\mu} + \frac{\partial \Phi}{\partial z}^T\mu^* + \frac{\partial \theta}{\partial z}^T\mu' - \frac{\partial L}{\partial z}^T = 0 \\
&-\bar{I}^T\dot{\mu}' + \frac{\partial f}{\partial \ell}^T\bar{\mu} + \frac{\partial \theta}{\partial \ell}^T\mu' - \frac{\partial L}{\partial \ell}^T = 0 \\
&\frac{\partial \Phi}{\partial z}\,\bar{\mu} - \frac{\partial L}{\partial \lambda}^T = 0
\end{aligned}\right\} \tag{4.9}$$

and terminal conditions

$$\bar{\mu}(\tau) = 0, \mu'(\tau) = 0, \mu^*(\tau) = 0 \tag{4.10}$$

The funcational $\delta\psi$ may then be rewritten as

$$\delta\psi = \Lambda^T \delta b \tag{4.11}$$

where

$$\Lambda^T = \frac{\partial g}{\partial b} + \bar{\mu}^T(0)P(b)\frac{\partial \nu}{\partial b} + \sum_{j=0}^{s-1} \mu'_{1+4j}(0)\frac{\partial \bar{\nu}_{j+1}}{\partial b} + \int_0^\tau \left[\frac{\partial L}{\partial b} - \bar{\mu}^T \frac{\partial (P(b)\dot{z})}{\partial b} - \bar{\mu}^T \frac{\partial f}{\partial b} - \mu^{*T}\frac{\partial \phi}{\partial b} - \mu'^T \frac{\partial \theta}{\partial b}\right] dt \tag{4.12}$$

One may now integrate the differential equations 4.9 from τ to 0, using the terminal conditions of Eq. 4.10. If one uses the implicit numerical algorithm of Eq. 3.8 for the above set of equations, the Newton iteration formula is

$$\begin{bmatrix} \left(\frac{-1}{h\beta_0} P^T + \frac{\partial f}{\partial z}^T\right) & \frac{\partial \theta}{\partial z}^T & \frac{\partial \phi}{\partial z}^T \\ \frac{\partial f}{\partial \ell}^T & \left(\frac{-1}{h\beta_0}\bar{I}^T + \frac{\partial \theta}{\partial \ell}^T\right) & 0 \\ \frac{\partial \phi}{\partial z} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\bar{\mu} \\ \Delta\mu' \\ \Delta\mu^* \end{bmatrix} = -\bar{g} \tag{4.13}$$

where $\bar{g}$ represents the left hand side of Eq. 4.9. For ready reference, it is desirable to repeat Eq. 3.8 applied to Eqs. 4.1 to 4.3 here as

$$\begin{bmatrix} \left(\frac{1}{h\beta_0} P + \frac{\partial f}{\partial z}\right) & \frac{\partial f}{\partial \ell} & \frac{\partial \phi}{\partial z}^T \\ \frac{\partial \theta}{\partial z} & \left(\frac{1}{h\beta_0}\bar{I} + \frac{\partial \theta}{\partial \ell}\right) & 0 \\ \frac{\partial \phi}{\partial z} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta \ell \\ \Delta \lambda \end{bmatrix} = - g \qquad (4.14)$$

where g represents the left-hand side of Eqs. 4.1 to 4.3.

Note that the coefficient matrix in Eq. 4.13 is exactly the transpose of the matrix of Eq. 4.14 with the exception of the $(-1/h\beta_0)$ term. But since one integrates from τ to 0, this implies negative time steps h, hence one need not modify the expressions $(1/h\beta_0 \; P + \frac{\partial f}{\partial z})$ and $(1/h\beta_0 \; \bar{I} + \frac{\partial \theta}{\partial \ell})$ of Eq. 4.14 for use in Eq. 4.13. Thus, one can use the same sparse matrix factorization code generated for dynamic analysis to solve Eq. 4.13 during adjoint calculations. The numerical efficiencies are obvious. This entire subject is addressed in detail in Ref. 9.

Method of Generalized Steepest Descent: The linearized optimal design problem is to find a vector δb that minimizes

$$\delta\tilde{\psi}_0 = \Lambda^{0^T} \delta b \qquad (4.15)$$

subject to design constraints

$$\delta\tilde{\psi} = \tilde{\Lambda}^T \delta b \quad \begin{cases} = - \psi_\beta, \; \beta=1,\ldots,r' \\ \leq - \psi_\beta, \; \beta\varepsilon(r'+1,\ldots,r), \; \psi_\beta \geq - \epsilon \end{cases} \qquad (4.16)$$

and the quadratic step-size constraint

$$\delta b^T W \delta b \leq \xi^2 \qquad (4.17)$$

where $\tilde{\psi}$ is a vector of ε-active constraints, $\tilde{\Lambda}$ contains sensitivity coefficients associated with ε-active constraints, W is a positive definite weighting matrix, and ξ and ε are small positive numbers.

Employing the Kuhn-Tucker necessary conditions of nonlinear programming [11], one derives the necessary design change δb as

$$\delta b = -\frac{1}{2\gamma_0}\,\delta b^1 + \delta b^2 \tag{4.18}$$

where

$$\delta b^1 = W^{-1}(\Lambda^0 + \tilde{\Lambda}\,\gamma^1) \tag{4.19}$$

$$\delta b^2 = -W^{-1}\,\tilde{\Lambda}\;\;\gamma^2 \tag{4.20}$$

$$M_{\psi\psi}\;\gamma^1 = -\tilde{\Lambda}^T\,W^{-1}\,\Lambda^0 \tag{4.21}$$

$$M_{\psi\psi}\;\gamma^2 = -\,\tilde{\psi} \tag{4.22}$$

$$M_{\psi\psi} = \tilde{\Lambda}^T\,W^{-1}\,\tilde{\Lambda} \tag{4.23}$$

and $\frac{1}{2\gamma_0} > 0$ is to be chosen as a step-size. The vector δb^1 is a constrained design derivative and δb^2 is a vector design change that corrects constraint errors.

5. Dynamic Analysis and Design System (DADS)

The DADS computer program implements the dynamic analysis, sensitivity analysis, and optimal design methods presented in Sections 2 to 4. Figures 5.1 and 5.2 are diagrams showing the subprograms that are used for dynamic and sensitivity analysis. Figure 5.3 gives the overall program flow diagram that incorporates these subprograms.

The dynamic analysis phase (DYNANL) of the program generates sparse matrix code for pivoting and LU factorization and solves the system of differential - algebraic equations for the state variables during a specified time interval. It employs sparse matrix codes and the numerical integration algorithm of Section 3. In addition, the Jacobian matrix and state variables at each time step are stored on a direct access disk file for subsequent use in sensitivity analysis.

The sensitivity analysis and optimization phase (SENANL) of the program carries out the calculations of Section 4. The program solves the system of adjoint equations using the previously stored data and the same numerical integration routine as above. The program further computes the sensitivity coefficients and necessary design improvements. This process is repeated until an optimum design is achieved.

The Dynamic Analysis Phase: The DYNANL phase of DADS establishes the sparse matrix code description of the mechanical system and numerically solves the differential and algebraic equations for the state variables. As shown in Fig. 5.1, this involves two major steps: (i) generation of an initial sparse code description (including pivoting and LU factorization code) and (ii) repetitive solution of the Newton iteration equations for the state variables during the time interval of interest.

In the first step of DYNANL, estimates of the initial configuration of the system are provided by INDATA and used by VARSET to initialize a state variable vector for subsequent use by the numerical integration routine DIFSUB. A compact numbering system identifying bodies, joints, and spring-damper elements is used to input data through INDATA and provides the necessary description of the mechanical system configuration. This information is used to construct the Newton corrector equations, Eqs. 4.14 and adjoint equations, Eqs. 4.13, for use in the sensitivity analsyis phase. The components of these equations are obtained from the equations of motion, Eqs. 2.23 and 2.24, spring-damper equations, Eqs. 2.11 and 2.13 to 2.16, and constraint equations, Eqs. 2.7 to 2.9. The program then uses subroutine S03000 to generate initial vectors of row and column indices that locate nonzero entries in the Jacobian matrix. Similarly, user supplied row-column positions of nonstandard elements are provided by incorporating the necessary code in USET. A symbolic description of the resultant matrix is printed by DEBUGG for reference purposes and a column ordering permutation vector is generated in S08000 (Section 3).

Subroutine S01000 evaluates coefficients and the right hand side of Eq. 3.8, or equivalently Eq. 4.14. Its functions are to (i) evaluate force and displacement functions of time that are provided by the user (FOREXT), (ii) transfer the state variables (Section 2) from a single vector used by the numerical integration routine to the standard variables and user-supplied variables (USOLV1), (iii) evaluate the Jacobian matrix for the standard equations and user-supplied equations (USOLV2) using updated variables from step (ii) and the previously generated (S08000) column ordering permutation vector, and (iv) evaluate the standard

equations and user-supplied equations (USOLV3). Finally a sparse LU factored description of the matrix is generated in INVERT.

The second step of DYNANL is to numerically integrate the system of equations during the time interval of interest. This is accomplished by the numerical integration subroutine DIFSUB, which repeatedly calls S01000 to update the Jacobian matrix and system of equations as it executes the iterative corrector formula of Eq. 4.14.

The Sensitivity Analysis Phase: In the SENANL phase, DYNANL is again called for adjoint calculation by AHLPSI, where the design sensitivity coefficients are calculated. For this case, DYNANL reads the previously generated data from disk file and executes its two step procedure. In the first step, sparse matrix codes for pivoting and LU factorization are generated for the transpose of the Jacobian matrix, using INVERT. In the second step, the system of adjoint equations is numerically integrated by DIFSUB. The program repeatedly reads data from the disk, calls S01NEW for reevaluation of the right hand side of the adjoint system of equations and iteratively solves for the adjoint variables at each time step. The adjoint variables are then stored on disk for later use in calculation of the sensitivity coefficients.

Description of the DADS Program: The Dynamic Analysis and Design System incorporates the two phases of Sections 5.2 and 5.3 into a design program that is capable of handling a variety of planar design problems. A brief description of how these phases are coupled as a design tool is given below. For more detailed descriptions, reference 9 is recommended.

Initial estimates and bounds on design paramters, the number of constraints and other parameters related to the design problem (Section 4) are read by the main program. Then system parameters such as masses, moments of inertia, location of centers of mass, applied constant forces, joint types, and spring-damper parameters are read through subroutine INDATA. Subroutine RELATE relates the variables or other system parameters of the dynamic analysis phase DYNANL (Section 2) to the updated design parameters. For example, if the location of a revolute joint on a link is changed, it may be necessary to change its mass and moment of inertia. This step is required following each design iteration. In the subsequent dynamic analysis phase the state equations are solved by DYNANL and the Jacobian matrix, state variables, time step, current time, and order of the numerical integration algorithm are stored on a direct access disk. This information is first used to evaluate the integrands of the cost and constraint functionals of Eqs. 4.4 and 4.5 and the integral form of Eq. 4.6 through subroutine HALS.

The inequality functional constraints of Eqs. 4.5 and 4.5 are then tested for violation and the corresponding adjoint equations of Eqs. 4.9 and 4.10 are solved (TEQFUN, INFUNC, ADJONT). The sensitivity coefficient matrix of Eq. 4.12 is calculated for each ε-active functional constraint (AHLPSI). Subroutine DPARMC tests each of the design parameter constraint and calculates the corresponding sensitivity matrices. The sensitivity vector Λ^0 of Eq. 4.15 is evaluated by AHLPSI. Subroutine NEWB computes the matrix $M_{\psi\psi}$ of Eq. 4.23, solves Eq. 4.21 and 4.22 for γ^1 and γ^2, and computes the necessary design changes δb given by Eqs. 4.18 to 4.20.

At this stage the convergence criteria are tested. If they are satisfied, the new design is taken as the optimum. Otherwise, the process is repeated with the new design parameters as the initial estimate.

Brief descriptions are given for other important subprograms appearing in Fig. 5.2:

AHSM: The function AHSM determines the maxima of the integrands of the transformed equality constraints obtained from Eq. 4.6. These maxima are used in TEQFUN.

FUNPSE: The function FUNPSE evaluates the cost and constraint functionals of Eqs. 4.4 and 4.5.

PBFN: The subroutine PBFN calculates terms of the matrix P(b) appearing in Eq. 4.1.

ADPDB: The subroutine ADPDB calculates derivatives of the elements of P(b) with respect to the design parameters b.

DGDBZ: The subroutine DGDBZ calculates the derivatives of the non integral parts of the cost and constraint functionals of Eqs. 4.4 and 4.5.

DNUDB: The subroutine DNUDB is used only when initial conditions of the dynamic analysis are dependent on design parameters. It evaluates the derivatives of the initial values of the solution variables with respect to design parameters.

ADLDZ: The subroutine ADLDZ calculates the derivatives $\frac{\partial L}{\partial z}$, $\frac{\partial L}{\partial \lambda}$, and $\frac{\partial L}{\partial \ell}$ in Eq. (4.9).

6. Applications and Numerical Results

The DADS program has been developed to treat analysis and design of quite general planar dynamic systems. To test the program a relatively simple slider crack mechanism is analyzed, design sensitivity analysis is carried out and the design is optimized.

The radial slider-crank mechanism is a four-bar linkage of rigid bodies that move in a plane. Fig. 6.1 shows the approximate initial position of such a mechanism with one spring-damper pair. Link 1 is ground, link 2 is the crank shaft, link 3 is the connecting rod or coupler, and link 4 is the piston or slider. A spring-damper pair is attached between link 4 and ground, as shown in Fig. 6.1. Revolute joints connect bodies 1 and 2, 2 and 3, 3 and 4. A translational joint connects bodies 4 and 1. Gravitational forces are ignored in the present simulation.

A symbolic listing of the nonzero positions of the Jacobian matrix for this example is given in Fig. 6.2. The only significance to be attached to the digits and letters is that there is a nonzero entry in each of the noted positions. All other entries are zeros, so only 9.5% of the matrix elements are nonzero and are accounted for by sparse matrix methods.

Formulation of the Optimal Design Problem: By virtue of its movement, a radial slider-crank mechanism exerts a force on ground through the crank bearing and the wrist-pin guide. It is desirable to keep these "shaking forces" within bounds. It is also desired that an upper bound be placed on the angular velocity of the crank at the final instant τ of the time-interval $[0,\tau]$ under consideration. The cost function is chosen

to be twice the maximum energy that is stored in the spring during the interval of motion. The design parameters shown in Fig. 6.1, are as follows:

b_1 = spring constant k_1 of the spring

b_2 = height of the points of attachment of the spring

b_3 = half of the length of the coupler

With the notation of Sections 2 and 4, the optimal design problem is stated as follows: Minimize

$$\psi_0 \equiv \max_{0 \le t \le \tau} b_1(\ell_1 - \ell_{10})^2 \tag{6.1}$$

subject to the equations of motion, Eqs. 2.23 and 2.24, the equations of constraint, Eqs. 2.6 to 2.9, the spring-damper relations, Eqs. 2.13 to 2.16, the initial conditions of the form of the second equation of Eqs. 4.2, the functional constraints

$$|\lambda_2^{12}| \le \lambda_2^{12}(\max), \qquad 0 \le t \le \tau \tag{6.2}$$

$$\dot{\phi}_2(\tau) \le \dot{\phi}_2(\max) \tag{6.3}$$

and the design parameter constraints

$$b_i^L \le b_i \le b_i^U, \qquad i = 1,2,3 \tag{6.4}$$

Here ℓ_{10} is the underformed spring length, ℓ_1 is the current spring length, and $\dot{\phi}_2$ is the angular velocity of the crankshaft. From Eqs. 2.6 and 2.13 the expression $-\frac{\partial \phi^{12}}{2y_2}\ \lambda_2^{12} = \lambda_2^{12}$ is interpreted as the y-component of the generalized force at this joint. Therefore, λ_2^{12} is the y-component of reaction force on the crankshaft at the joint between the crankshaft

and ground. For the sake of simplicity, only the constraint on the vertical component of shaking force at the crank bearing is considered here.

After introduction of an artificial design parameter b_4 [11], and the integral functional forms in the conventional way [11], the problem can be reformulated as follows: Minimize

$$\psi_0 = b_4 \tag{6.5}$$

subject to Eqs. 2.23 to 2.24, 2.6 to 2.9, 2.13 to 2.16, and the second equations of Eqs. 4.1 and 4.2; the constraints

$$\psi_1 \equiv \int_0^\tau < -\lambda_2^{12} - \lambda_2^{12}(\max) > dt = 0 \tag{6.6}$$

$$\psi_2 \equiv \int_0^\tau < \lambda_2^{12} - \lambda_2^{12}(\max) > dt = 0 \tag{6.7}$$

$$\psi_3 \equiv \int_0^\tau < b_1(\ell_1 - \ell_{10})^2 - b_4 > dt = 0 \tag{6.8}$$

$$\psi_4 \equiv \dot{\phi}_2(\tau) - \dot{\phi}_2(\max) \leq 0 \tag{6.9}$$

and the design parameter constraints of Eq. 6.4. Here, the symbol $<\eta(t)>$ has the following meaning:

$$\langle \eta(t) \rangle = \begin{cases} (\eta(t))^2 , & \text{for } \eta(t) \geq 0 \\ 0 , & \text{for } \eta(t) < 0 \end{cases} \tag{6.10}$$

This formulation is a special case of the general formulation of the optimal design problem stated in Section 4. After specification of initial numerical data, the DADS program described in Section 5 can be implemented to obtain its solution.

Design Sensitivity Analysis: In optimal design, calculation of design derivatives constitutes a principal part of the work to be done. When design sensitivity coefficients are obtained, the gradient projection iterative optimization algorithm of Section 4 can be applied. In the present problem, subroutines RELATE, DNUDB, DGDBZ, ADLDZ, and DLDFB are the major user-supplied subprograms that are required for design sensitivity analysis. Among them, DGDBZ and ADLDZ are often simple, because they depend solely on the forms of the functional constraints. Subroutines DNUDB, DGDBZ, and DLDFB are used in AHLPSI to calculate sensitivity coefficients.

Numerical Results: Initial estimates of the parameters for the slider crank mechanism under consideration are given in Table 6.1. The table associates the computer code names with the corresponding parameters in Eqs. 2.4, 2.6 to 2.9, 2.13 to 2.16, and 2.23. Two simulation time intervals are considered; [0,1] and [0,2] seconds. During transient analysis a constant counter-clockwise torque of 100 in-lbf is applied to link 2.

To study the numerical results of design sensitivity analysis, the sensitivity coefficients $\ell^{\psi\beta}$, $\beta\varepsilon(1,2,3,4)$, and ℓ^{ψ_0} were first calculated

for the 2 second interval, with initial estimates of the design parameters given in Table 6.2a. Lower and upper bounds on the first three design parameters were selected as $b^L = [0.8, 0.2, 5.0]^T$ and $b^U = [1.5, 0.8, 16.0]$. Also $\lambda_2^{12}(\max) = 10.0$ and $\dot{\phi}_2(\max) = 0.3$.

The cost and constraint functional were evaluated for 0.1% and 1.0% perturbations of the first, second, and fourth design parameters and 0.01% and 0.1% changes in the third design parameter. The corresponding changes in the functionals given in Table 6.2a are presented in compact form. In these computations the integrands of ψ_1, ψ_2, and ψ_3 have been normalized by dividing by $\lambda_2^{12}(\max)$, $\lambda_2^{12}(\max)$, and b_4^2, respectively. Sensitivity coefficients are given in Table 6.2b.

The predicted changes $\delta\psi_\beta = \ell^{\psi\beta}\delta b$ in the cost and constraint functions due to the foregoing design perturbations were calculated and are given in Table 6.2a. It is observed from the table that terms of the form $\ell^{\psi\beta}\delta b$ match satisfactorily with the actual changes in ψ_β, $\beta = 0,3,4$. The slight discrepancies are attributed to various approximations and linearizations at many steps of both transient and sensitivity analysis and the coarse time grid used for numerical integration.

In carrying out the optimal design algorithm for the two second interval, a design reduction ratio of 3 percent was used to compute the step size in the first iteration. The initial estimate b^0 and design variable bounds are given in Table 6.3.

At the starting design $||\delta b^1|| = 0.1248$ and $||\delta b^2|| = 0.2307$. Constraints 3 and 4 are violated and the lower bound constraint on design parameter b_3 is tight. After 6 iterations, as the algorithm approached

the optimum design, $||\delta b^1||$ reduced to 0.8404×10^{-5}, $||\delta b^2||$ reduced to 0.2669×10^{-1}, and design variable constraints on b_1, b_2 and b_3 became tight. The optimum design is given in Table 6.3.

Next, the design problem is considered for the interval [0,1], with design and initial data given in Table 6.4. For this case a design reduction ratio of 10% was used in computing the step size in the first iteration. At the starting design $||\delta b^1|| = 0.0$, $||\delta b^2|| = 0.1106 \times 10^{-2}$, and constraint 3 is violated. In the second iteration $||\delta b^1|| = 0.7348$, $||\delta b^2|| = 3.814 \times 10^{-2}$, and again constraint 3 is violated. After 18 iterations, as the algorithm approaches the optimum design, $||\delta b^1|| = 5.567 \times 10^{-3}$, $||\delta b^2|| = 3.018 \times 10^{-3}$, and design parameter constraints on b_1 and b_2 are tight. The optimum design is given in Table 6.4.

References

1. Chua, L. O. and Lin, P-M., Computer Aided Analysis of Electronic Circuits, Prentice-Hall, Englewood Cliffs, New Jersey, 1975.

2. Zienkiewicz, D. C., The Finite Element Method, Third Edition, McGraw-Hill, New York, 1977.

3. Paul, B., "Analytical Dynamics of Mechanisms - A Computer Oriented Overview", Mechanism and Machine Theory, Vol. 10, 1975, pp. 481-507.

4. Sheth, P. N., "A Digital Computer Based Simulation Procedure for Multiple Degree of Freedom Mechanical Systems with Geometric Contraints," Ph.D. Thesis, University of Wisconsin, Madison, Wisconsin, 1972.

5. Chace, M. A. and Smith, D. A., "DAMN - Digital Computer Program for the Dynamic Analysis of Generalized Mechanical Systems", SAE Transactions, Vol. 80, 1971, pp. 969-987.

6. Orlandea, N., Node-Analogous, Sparsity Oriented Methods for Simulation of Mechanical Dynamic Systems," Ph.D. Thesis, University of Michigan, Ann Arbor, Michigan, 1973.

7. Greenwood, D. T., Principles of Dynamics, Prentice-Hall, Englewood Cliffs, New Jersey, 1965.

8. Brameller, A., Allen, R. N., and Haman, Y. M., Sparsity, Pitman Publishing Corp., New York, 1976.

9. Barman, N. C., Sensitivity Analysis and Optimization of Constrained Mechanical System Dynamics, Ph.D. Thesis, Materials Division, University of Iowa, 1978.

10. Haug, E.J., and Arora, J.S., "Design Sensitivity Analysis of Elastic Mechanical Systems", Computer Methods in Applied Mechanics and Engineering, Vol. , 1978, pp.

11. Haug, E.J., and Arora, J.S. Applied Optimal Design, John Wiley, New York, 1979, (to appear).

TABLE 6.1 Initial Estimates of the Parameters for the Slider-Crank Mechanism (Units in Inch, Pound-Force, Second System).

LINK DESCRIPTION

m_1 =M(1) =1.	m_2 =M(2) =4.	m_3 =M(3) =1.	m_4 =M(4) =3.
J_1 =JIN(1) =1.	J_2 =JIN(2) =10.	J_3 =JIN(3) =4.	J_4 =JIN(4) =2.
x_1 =X(1) =0.	x_2 =X(2) =1.	x_3 =X(3) =8.667	x_4 =X(4) =17.32
y_1 =Y(1) =0.	y_2 =Y(2) =0.	y_3 =Y(3) =5.25	y_4 =Y(4) =.5
ϕ_1 =PHI(1) =0.	ϕ_2 =PHI(2) =1.5708	ϕ_3 =PHI(3) =-.5236	ϕ_4 =PHI(4) =0.
* Q_{x_1} =FX(1) =0.	Q_{x_2} =FX(2) =0.	Q_{x_3} =FX(3) =0.	Q_{x_4} =FX(4) =0.
* Q_{y_1} =FY(1) =0.	Q_{y_2} =FY(2) =0.	Q_{y_3} =FY(3) =0.	Q_{y_4} =FY(4) =0.
* Q_{ϕ_1} =TQ(1) =0.	Q_{ϕ_2} =TQ(2) =100.	Q_{ϕ_3} =TQ(3) =0.	Q_{ϕ_4} =TQ(4) =0.

JOINT DESCRIPTION

i =IB(1,1) =1	i =IB(1,2)=2	i =IB(1,3)=3	i =IB(1,4) =4
ξ_{12} =X1(1) =0.	ξ_{23} =X1(2) =9.0	ξ_{34} =X1(3) =10.	ξ_{41}=X1(4) =0.
η_{12} =Y1(1) =0.	η_{23} =Y1(2) =0.	η_{34} =Y1(3) =0.	η_{41}=Y1(4) =0.5
j =IB(2,1) =2	j =IB(2,2)=3	j =IB(2,3)=4	j =IB(2,4) =1
ξ_{21} =X2(1) =-1.	ξ_{32} =X2(2) =-10.	ξ_{43} =X2(3) =0.	ξ_{14}=X2(4) =0.
η_{21} =Y2(1) =0.	η_{32} =Y2(2) =0.	η_{43} =Y2(3) =0.	η_{14}=Y2(4) 1.0

SPRING-DAMPER DESCRIPTION

i =IBSD(1,1)=1	$\xi_{s_{14}}$=XF1(1) =35.	$\eta_{s_{14}}$=YF1(1) =0.5	j =IBSD(2,1)=4
$\xi_{s_{41}}$=XF2(1) =2.	$\eta_{s_{41}}$=YF2(1) =0.	k_{14} =SK(1) =1.	C_{14}=DC(1) =1.
ℓ_{14} =SDL(1) =15.68	$\ell_{0_{14}}$=SDL0(1)=15.68	$F_{0_{14}}$=SDF(1) =0.	

* The parameters FX, FY, and TQ represent components of force and moment that are reduced to the centers of mass on each link and contribute to Q_{x_i}, Q_{y_i}, and Q_{ϕ_i} in Eq. 2.4.

TABLE 6.2a Sensitivity Analysis Results for the Slider-Drank Mechanism in Compact Form.

Initial estimate of the design parameters: $\underline{b}_0 = (1.0, 0.5, 10.0, 14.5)^T$

δb_1	δb_2	δb_3	δb_4	J	ΔJ	$\ell^J \delta b$	ψ_1	ψ_2
0.0	0.0	0.0	0.0	14.5				
0.1×10^{-2}	0.0	0.0	0.0	14.5	0.0	0.0		
0.0	0.5×10^{-3}	0.0	0.0	14.5	0.0	0.0	UNVIOLATED	UNVIOLATED
0.0	0.0	$.1 \times 10^{-2}$	0.0	14.5	0.0	0.0		
0.0	0.0	0.0	0.145×10^{-1}	14.5145	0.145×10^{-1}	0.145×10^{-1}		
0.1×10^{-1}	0.0	0.0	0.0	14.5	0.0	0.0		
0.0	0.5×10^{-2}	0.0	0.0	14.5	0.0	0.0		
0.0	0.0	0.1×10^{-1}	0.0	14.5	0.0	0.0		
0.0	0.0	0.0	0.145	14.645	0.145	0.145		

TABLE 6.2a (continued)

δb_1	δb_2	δb_3	δb_4	ψ_3	$\Delta\psi_3$	$\ell^{\psi 3}\ \delta b$	ψ_4	ψ_4	$\ell^{\psi 4}\ \delta b$
0.0	0.0	0.0	0.0	0.16601×10^{-3}			0.24144		
0.1×10^{-2}	0.0	0.0	0.0	0.17061×10^{-3}	0.0046×10^{-3}	0.0061×10^{-3}	0.24134	-0.0001	-0.00009
0.0	0.5×10^{-3}	0.0	0.0	0.16601×10^{-3}	0.0	0.0	0.24144	0.0	0.0
0.0	0.0	0.1×10^{-2}	0.0	0.16717×10^{-3}	0.1160×10^{-5}	0.1395×10^{-5}	0.24172	0.0002	0.0002
0.0	0.0	0.0	0.145×10^{-1}	0.16073×10^{-3}	-0.0053×10^{-3}	-0.0064×10^{-3}	0.24144	0.0	0.0
0.1×10^{-1}	0.0	0.0	0.0	0.21534×10^{-3}	0.04933×10^{-3}	0.06115×10^{-3}	0.24039	-0.0010	-0.0009
0.0	0.5×10^{-2}	0.0	0.0	0.16601×10^{-3}	0.0	0.0	0.24144	0.0	0.0
0.0	0.0	0.1×10^{-1}	0.0	0.17783×10^{-3}	0.01182×10^{-3}	0.01395×10^{-3}	0.24430	0.0028	0.0020
0.0	0.0	0.0	0.145	0.11825×10^{-3}	-0.0478×10^{-3}	-0.0647×10^{-3}	0.24144	0.0	0.0

TABLE 6.2b Sensitivity Coefficients ℓ^J and ℓ^{ψ_β}, $\beta \in \{3,4\}$.

Components	ℓ^J	ℓ^{ψ_3}	ℓ^{ψ_4}
1	0.0	0.6115×10^{-2}	-0.850×10^{-1}
2	0.0	0.0	0.0
3	0.0	0.1395×10^{-2}	0.1872
4	1.0	-0.4626×10^{-3}	0.0

TABLE 6.3 Optimum Results for the Slider Crank Mechanism for the Time Interval [0.0, 2.0].

Cost function $\psi_0 = \max_{0 \leq t \leq 2} b_1(\ell_1 - \ell_{10})^2$

Lower bounds on b, $b^L = [0.8, 0.2, 9.0]^T$

Upper bounds on b, $b^U = [1.5, 0.8, 12.0]^T$

	Starting Values	Optimum Values
b_1	8.2440×10^{-1}	8.0000×10^{-1}
b_2	5.0000×10^{-1}	5.0000×10^{-1}
b_3	9.0000	9.0000
ψ_0	1.1159×10^{1}	1.00115×10^{1}

TABLE 6.4 Optimum Results for the Slider Crank Mechanism for the Time Interval [0.0, 1.0].

Cost function $\psi_0 = \max_{0 \le t \le 1} b_1(\ell_1 - \ell_{10})^2$

Lower bounds on b, $b^L = [0.8, 0.2, 5.0]^T$

Upper bounds on b, $b^U = [1.5, 0.8, 16.0]^T$

	Starting Values	Optimum Values
b_1	1.0000	8.0000×10^{-1}
b_2	5.0000×10^{-1}	5.0000×10^{-1}
b_3	1.0000×10^{1}	9.7566
ψ_0	1.2417	9.5767×10^{-1}

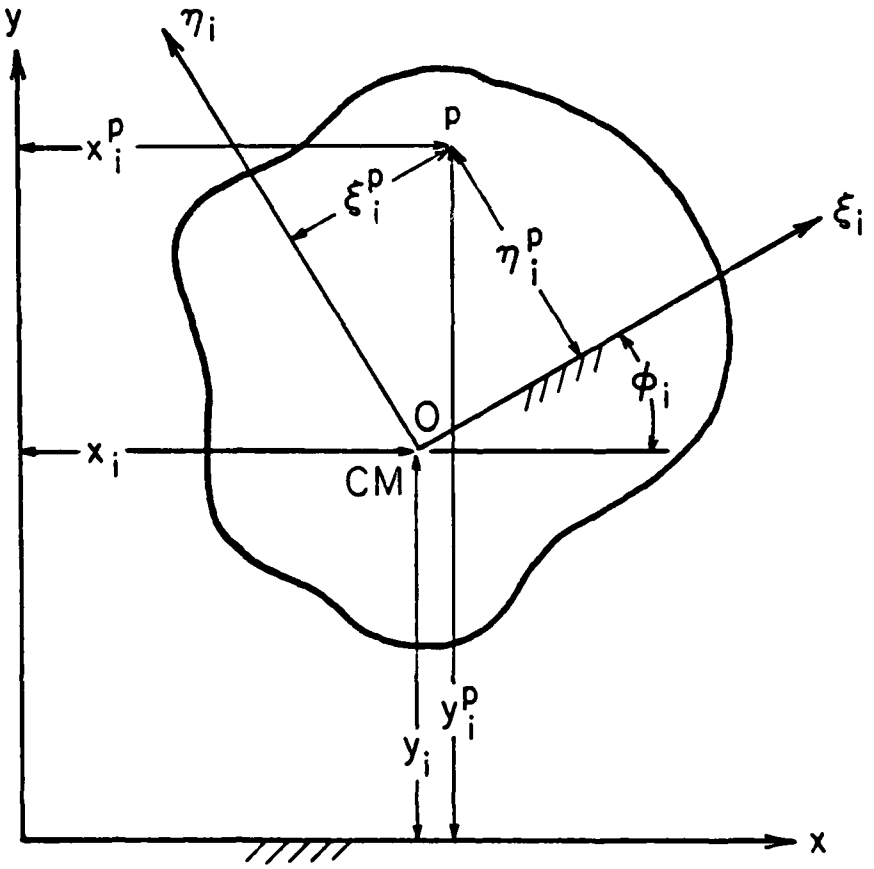

Figure 2.1 Definition of the Generalized Co-ordinate of the i-th Body.

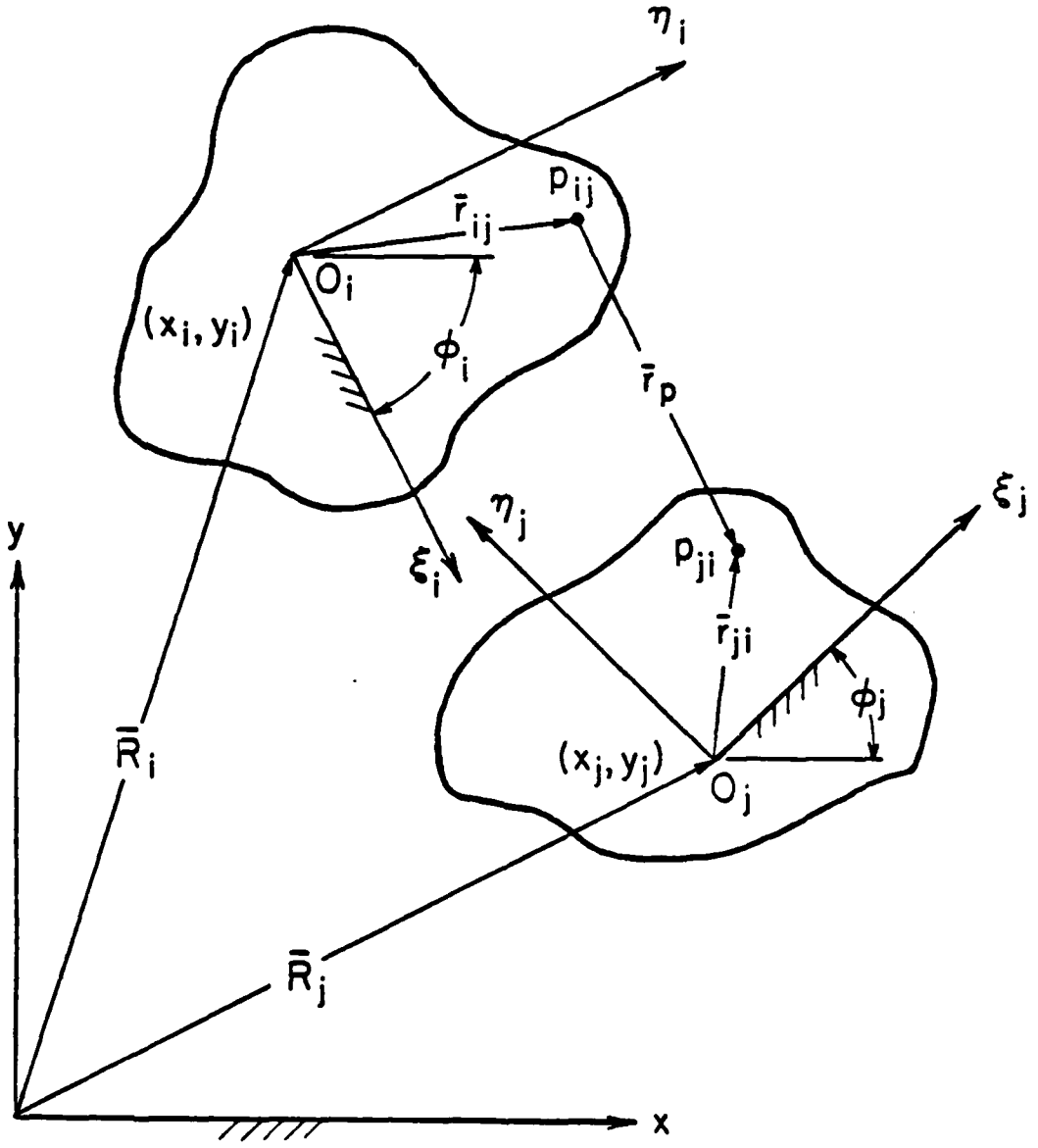

Figure 2.2 Joint Co-ordinates

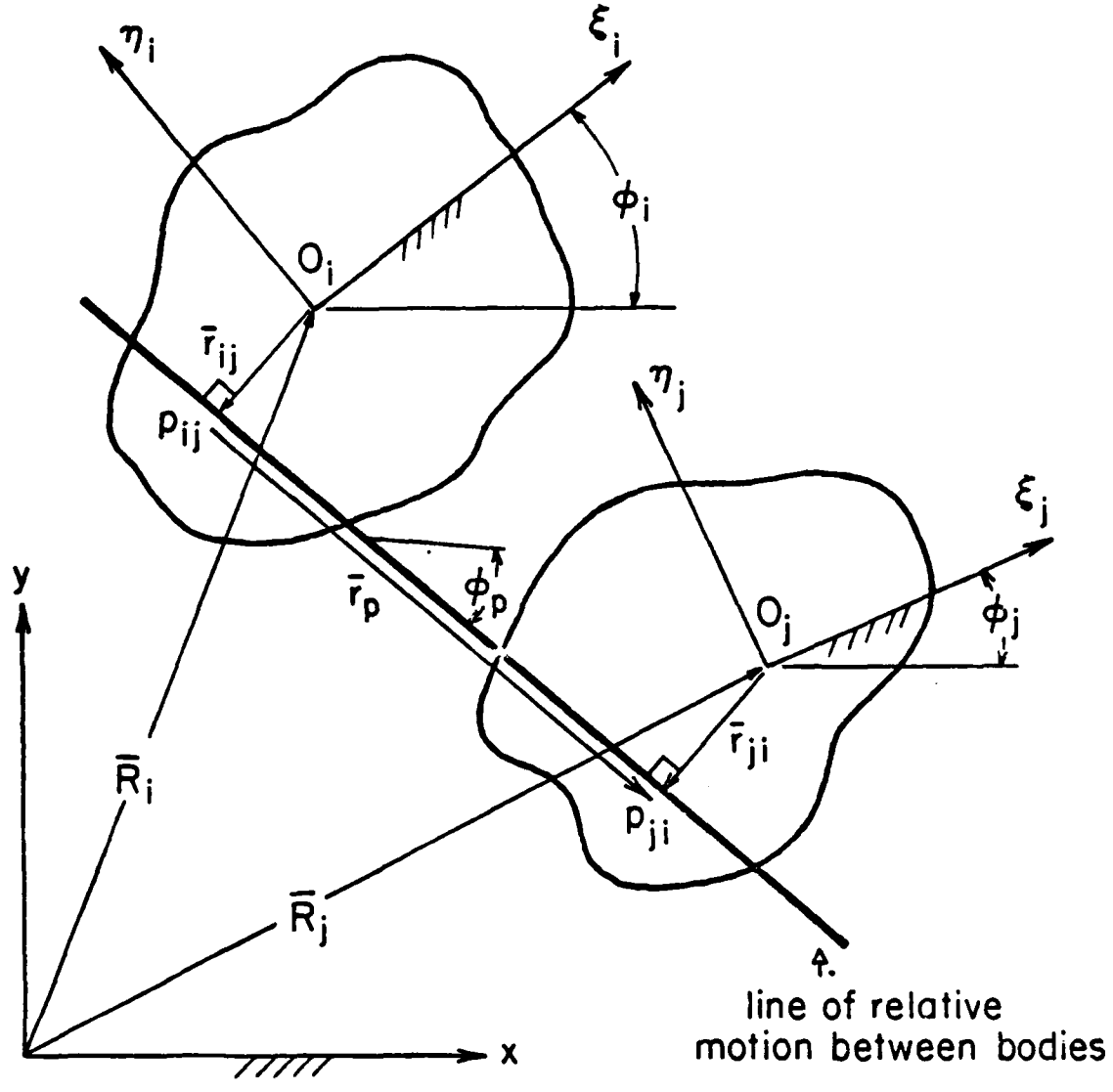

Figure 2.3 Translational Joint

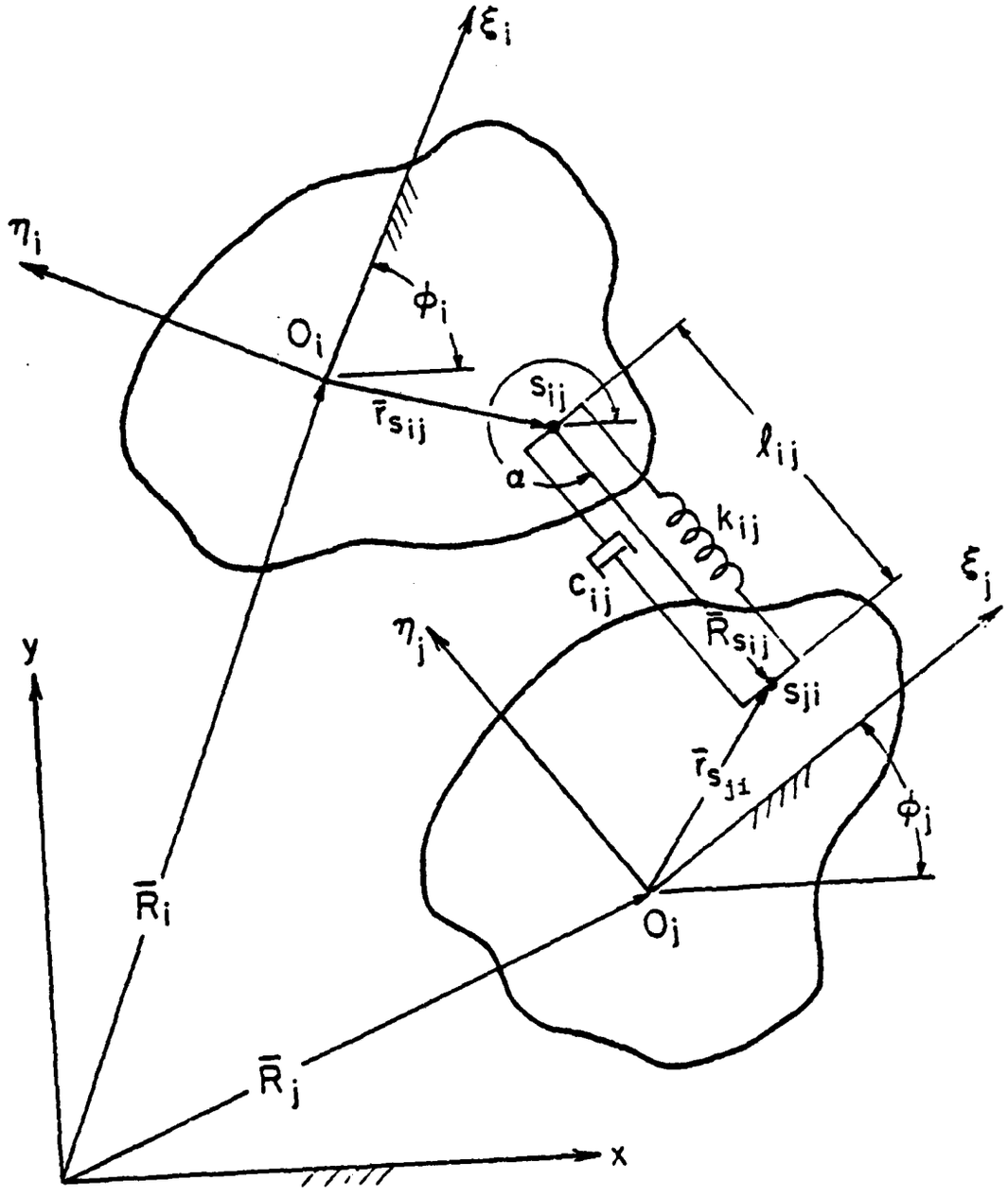

Figure 2.4 Co-ordinate for the Spring-Damper Combination

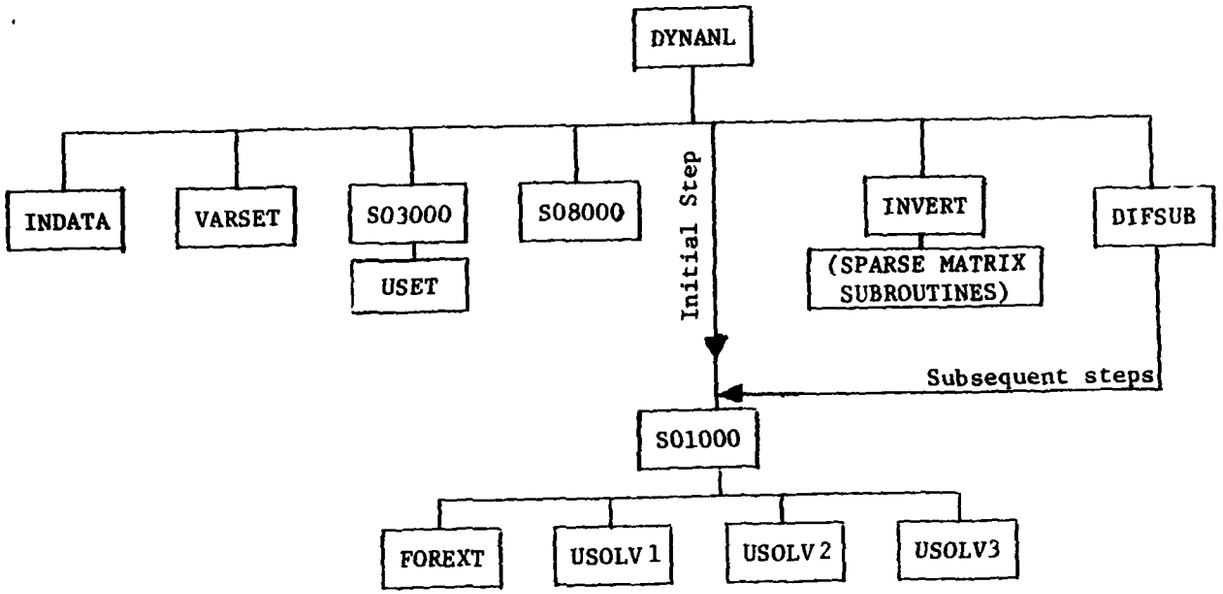

Figure 5.1 DADS - Dynamic Analysis Phase.

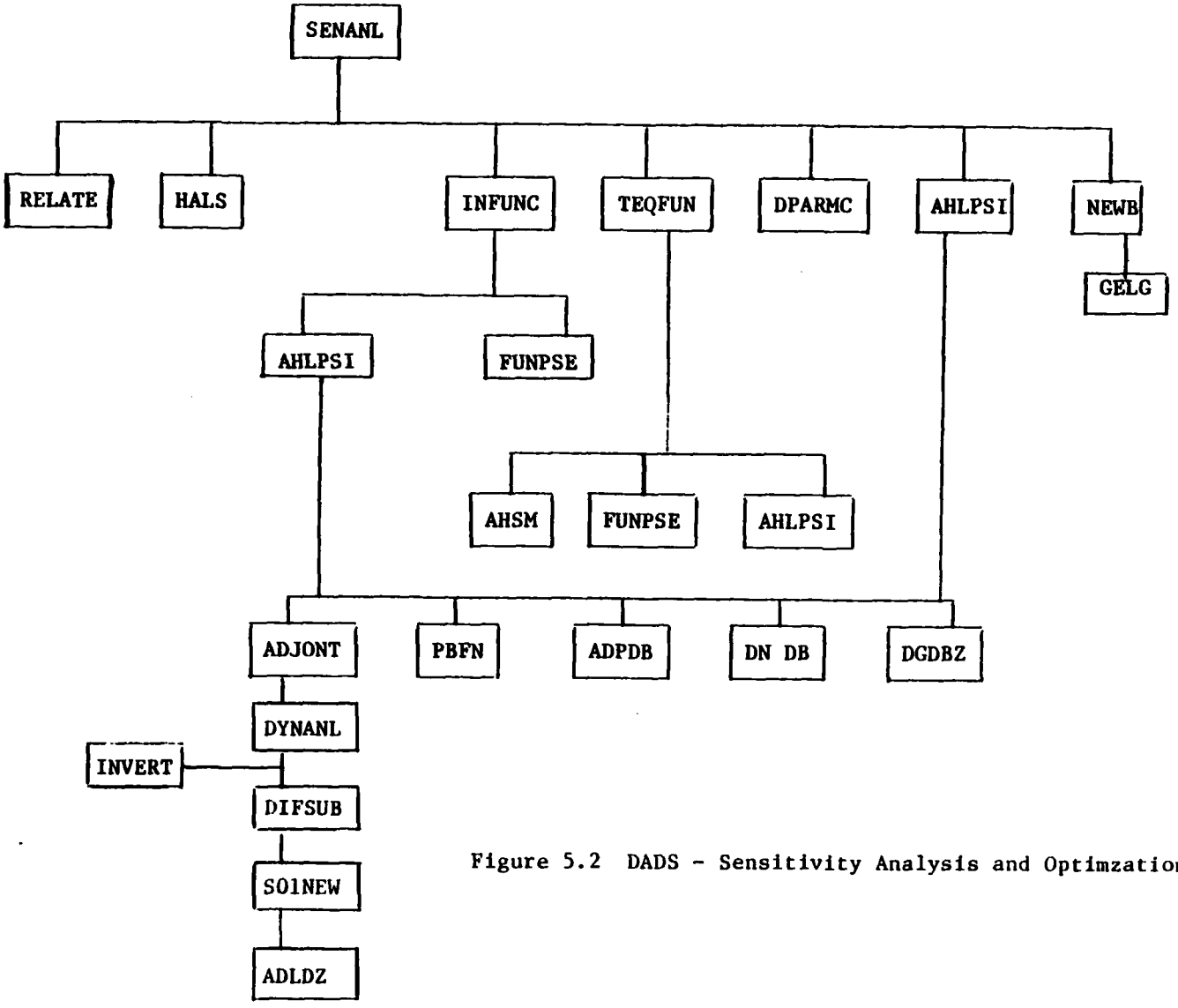

Figure 5.2 DADS - Sensitivity Analysis and Optimzation Phase

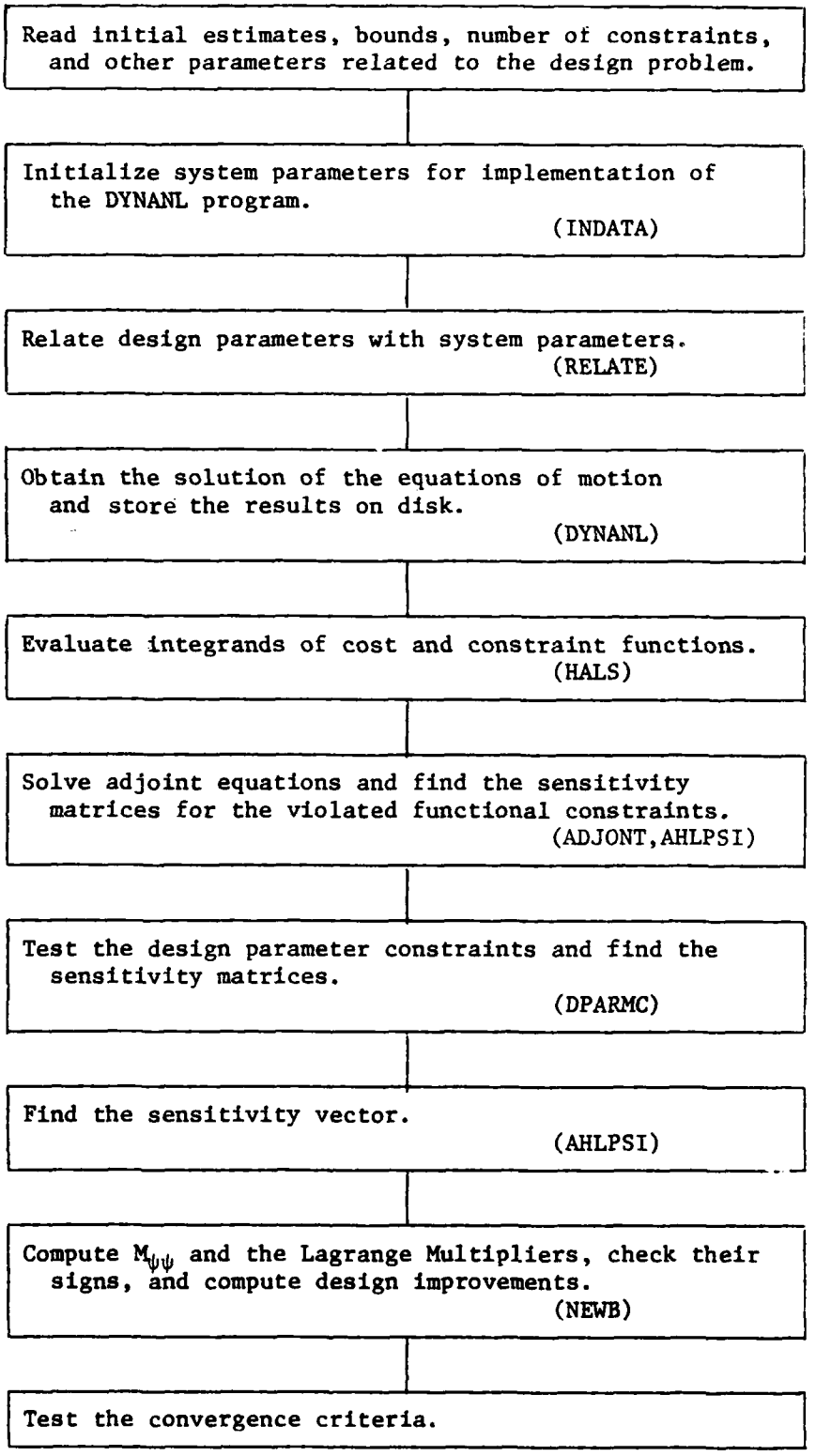

Figure 5.3 Flow Diagram of the DADS Computer Program.

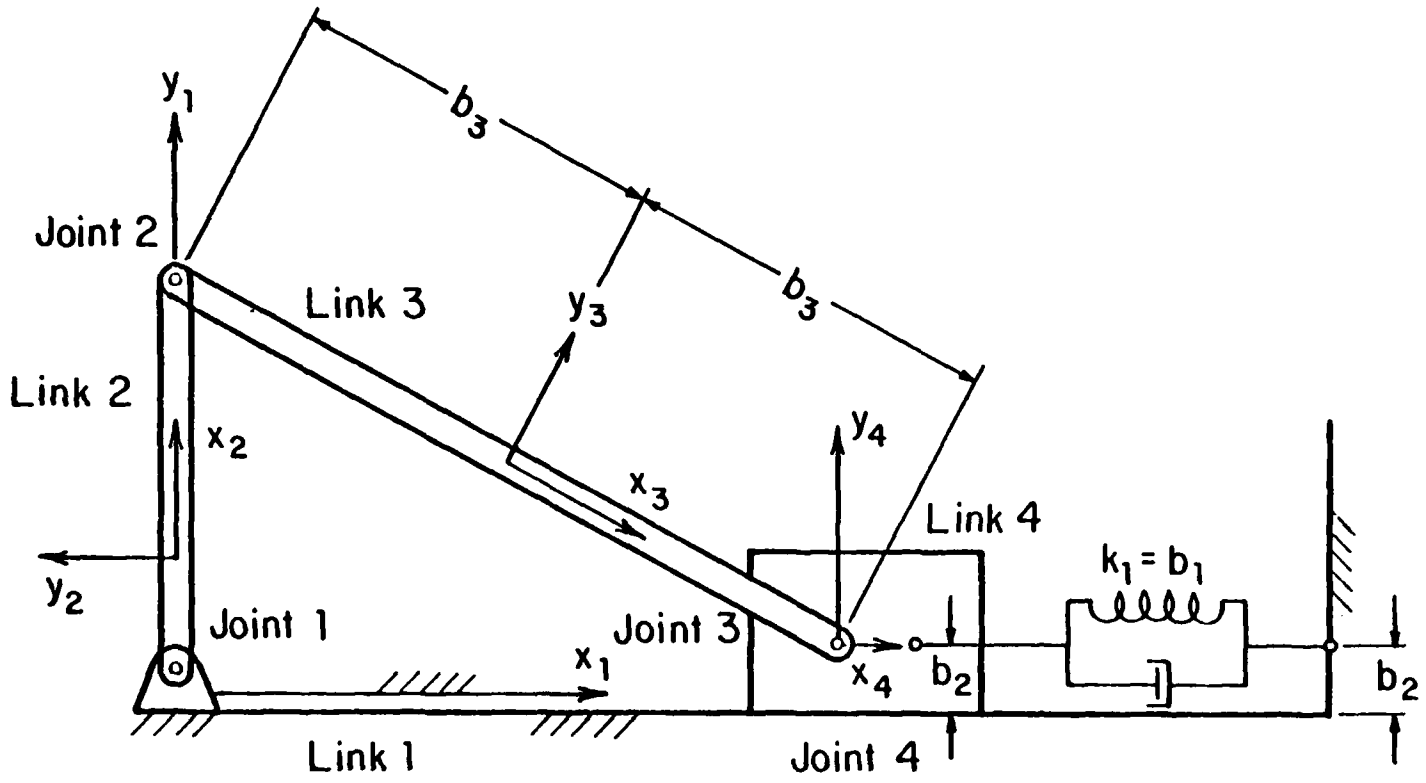

Figure 6.1 Approximate Initial Configuration of the Slider-Crank Mechanism

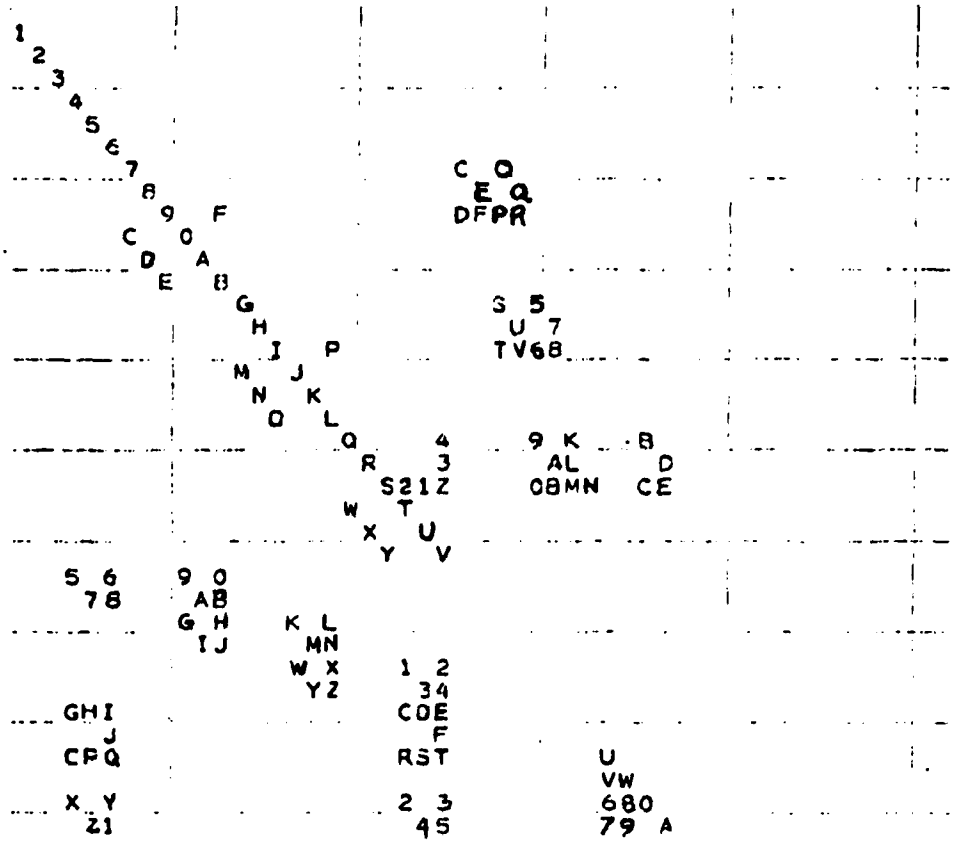

Figure 6.2 Symbolic Listing of the Nonzero Entries in the Jacobian Matrix for the Example Slider-Crank Mechanism

DISTRIBUTION LIST

Please notify USATACOM, DRSTA-ZSA, Warren, Michigan 48090, of corrections and/or changes in address.

Commander (25)
US Army Tk-Autmv Command
R&D Center
Warren, MI 48090

Superintendent (02)
US Military Academy
ATTN: Dept of Engineering
Course Director for
Automotive Engineering

Commander (01)
US Army Logistic Center
Fort Lee, VA 23801

US Army Research Office (02)
P.O. Box 12211
ATTN: Dr. F. Schmiedeshoff
Dr. R. Singleton
Research Triangle Park, NC 27709

HQ, DA (01)
ATTN: DAMA-AR
Dr. Herschner
Washington, D.C. 20310

HQ, DA (01)
Office of Dep Chief of Staff
for Rsch, Dev & Acquisition
ATTN: DAMA-AR
Dr. Charles Church
Washington, D.C. 20310

HQ, DARCOM
5001 Eisenhower Ave.
ATTN: DRCDE
Dr. R.L. Haley
Alexandria, VA 22333

Director (01)
Defense Advanced Research
Projects Agency
1400 Wilson Boulevard
Arlington, VA 22209

Commander (01)
US Army Combined Arms Combat
Developments Activity
ATTN: ATCA-CCC-S
Fort Leavenworth, KA 66027

Commander (01)
US Army Mobility Equipment
Research and Development Command
ATTN: DRDME-RT
Fort Belvoir, VA 22060

Director (02)
US Army Corps of Engineers
Waterways Experiment Station
P.O. Box 631
Vicksburg, MS 39180

Commander (01)
US Army Materials and Mechanics
Research Center
ATTN: Mr. Adachi
Watertown, MA 02172

Director (03)
US Army Corps of Engineers
Waterways Experiment Station
P.O. Box 631
ATTN: Mr. Nuttall
Vicksburg, MS 39180

Director (04)
US Army Cold Regions Research
& Engineering Lab
P.O. Box 282
ATTN: Dr. Freitag, Dr. W. Harrison
Dr. Liston, Library
Hanover, NH 03755

President (02)
Army Armor and Engineer Board
Fort Knox, KY 40121

Commander (01)
US Army Arctic Test Center
APO 409
Seattle, WA 98733

Commander (02)
US Army Test & Evaluation
Command
ATTN: AMSTE-BB and AMSTE-TA
Aberdeen Proving Ground, MD
21005

Commander (01)
US Army Armament Research
and Development Command
ATTN: Mr. Rubin
Dover, NJ 07801

Commander (01)
US Army Yuma Proving Ground
ATTN: STEYP-RPT
Yuma, AZ 85364

Commander (01)
US Army Natic Laboratories
ATTN: Technical Library
Natick, MA 01760

Director (01)
US Army Human Engineering Lab
ATTN: Mr. Eckels
Aberdeen Proving Ground, MD
21005

Director (02)
US Army Ballistic Research Lab
Aberdeen Proving Ground, MD
21005

Director (02)
US Army Materiel Systems
Analysis Agency
ATTN: AMXSY-CM
Aberdeen Proving Ground, MD
21005

Director (02)
Defense Documentation Center
Cameron Station
Alexandria, VA 22314

US Marine Corps (01)
Mobility & Logistics Division
Development and Ed Command
ATTN: Mr. Hickson
Quantico, VA 22134

Keweenaw Field Station (01)
Keweenaw Research Center
Rural Route 1
P.O. Box 94-D
ATTN: Dr. Sung M. Lee
Calumet, MI 49913

Naval Ship Research & (02)
Dev Center
Aviation & Surface Effects Dept
Code 161
Washington, D.C. 20034

Director (01)
National Tillage Machinery Lab
Box 792
Auburn, AL 36830

Director (02)
USDA Forest Service Equipment
Development Center
444 East Bonita Avenue
San Dimes, CA 91773

Engineering Societies (01)
Library
345 East 47th Street
New York, NY 10017

Dr. I.R. Erlich (01)
Dean for Research
Stevens Institute of Technology
Castle Point Station
Hoboken, NJ 07030

Grumman Aerospace Corp (02)
South Oyster Bay Road
ATTN: Dr. L. Karafiath
Mr. F. Markow
M/S A08/35
Bethpage, NY 11714

Dr. Bruce Liljedahl (01)
Agricultural Engineering Dept
Purdue University
Lafayette, IN 46207

Mr. H.C. Hodges (01)
Nevada Automotive Test Center
Box 234
Carson City, NV 89701

Mr. R.S. Wismer (01)
Deere & Company
Engineering Research
3300 River Drive
Moline, IL 61265

Oregon State University (01)
Library
Corvallis, OR 97331

Southwest Research Inst (01)
8500 Culebra Road
San Antonia, TX 78228

FMC Corporation (01)
Technical Library
P.O. Box 1201
San Jose, CA 95108

Mr. J. Appelblatt (01)
Director of Engineering
Cadillac Gauge Company
P.O. Box 1027
Warren, MI 48090

Chrysler Corporation (02)
Mobility Research Laboratory,
Defense Engineering
Department 6100
P.O. Box 751
Detroit, MI 48231

CALSPAN Corporation (01)
Box 235
Library
4455 Benesse Street
Buffalo, NY 14221

SEM, (01)
Forsvaretsforskningsanstalt
Avd 2
Stockholm 80, Sweden

Mr. Hedwig (02)
RU III/6
Ministry of Defense
5300 Bonn, Germany

Foreign Science & Tech (01)
Center
220 7th Street North East
ATTN: AMXST-GEI
Mr. Tim Nix
Charlottesville, VA 22901

General Research Corp (01)
7655 Old Springhouse Road
Westgate Research Park
ATTN: Mr. A. Viilu
McLean, VA 22101

Commander (01)
US Army Developmant and
Readiness Command
5001 Eisenhower Avenue
ATTN: Dr. R.S. Wiseman
Alexandria, VA 22333